

第五章

軟體專案範疇管理

執行單位：國立臺灣科技大學
軟體工程學程中心

大綱

- 軟體專案評選
- 軟體專案範疇定義
- 軟體專案範疇確認
- 軟體專案範疇控制
- 軟體專案範疇變更

軟體專案評選

■ 軟體專案評選 (Software Project Evaluation and Selection)

- 基本概念
- 決策問題與過程。
- 專案選擇方法

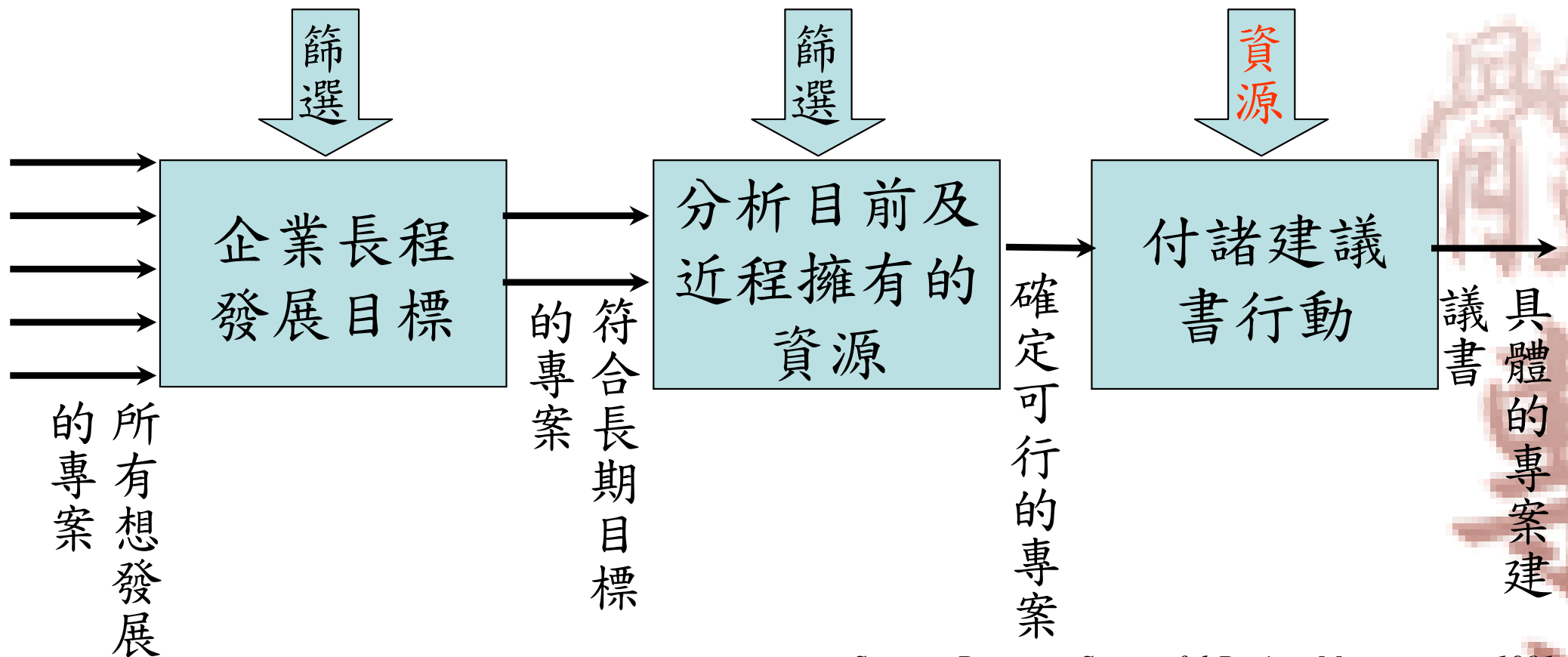
專案評選的基本概念 (1/2)

- 所謂「專案評選」乃是對若干個別的專案或一群專案先進行評估，再選擇其中一部份來執行，使其能符合組織的發展目標，並可落實組織的策略的一種程序。
- 為使該項作業能有效實施，通常是需運用一套系統化的程序—在有限的資源情況下，下達決策以將相互競爭的備選方案中擇其最佳者來執行。
- 當一項專案被選擇出即表示對它「未來的承諾」，乃是承諾給予：適人、適職、經費及資源。

專案評選的基本概念 (2/2)

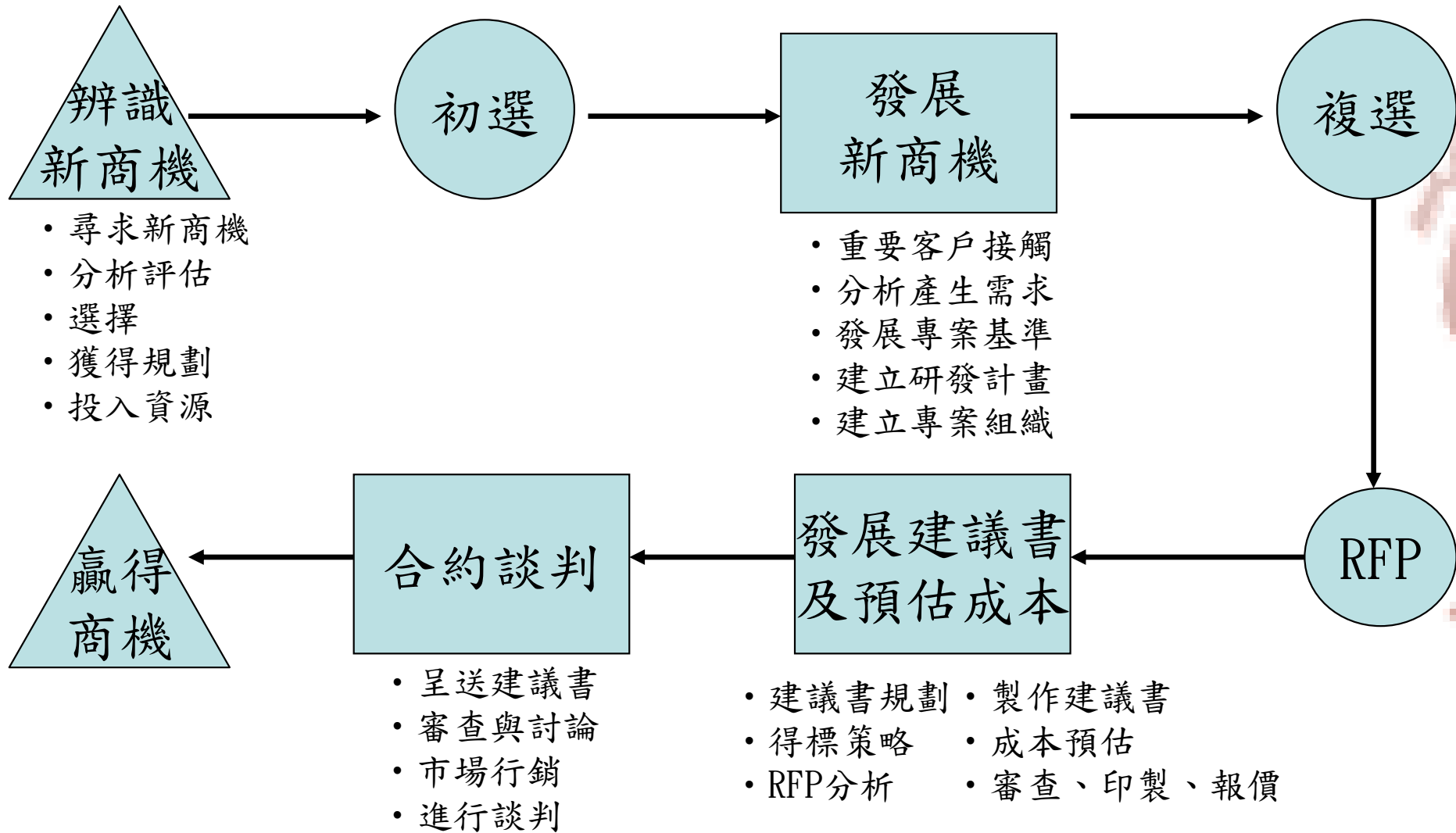
- 若「專案評選」的程序是非常的粗糙草率或設計不當，組織就必須付出一些代價及損失，並可能影響未來長遠的發展。
- 在「專案評選」程序中基本上最重要的是必須要能明確瞭解「需要—需求生命週期（needs-requirements lifecycle）」的意涵，並將其納入考量。

需要—評估—選擇程序 (The Needs-Evaluation Selection Process) : 專案選擇簡易法之一



Source: Rosenau, *Successful Project Management*, 1981

需要－評估－選擇程序 (The Needs-Evaluation Selection Process) : 專案選擇簡易法之二



專案管理中之決策問題 (1/2)

- 制訂決策為專案中不論主管、專案經理或是專案人員均須面對之問題。
- 專案過程中可能面對之時機包含：
 - 企業必須決定多專案之優先順序以利資源分配。
 - 面對環境變化時，專案執行中有關go/no-go之問題。
 - 專案人員提出變更需求時之取捨決定。
 - 採購、風險評估及因應、評估控管時各項決定。

專案管理中之決策問題 (2/2)

- 決策之正確與否將決定專案之品質及專案產品之品質，專案人員必須慎選決策工具及理則以避免影響專案，進而影響企業之獲利。

直覺、經驗及系統思考

直覺

- 由於人有求生存之本能，面對問題時自然產生因應能力，以主觀判定解決問題。

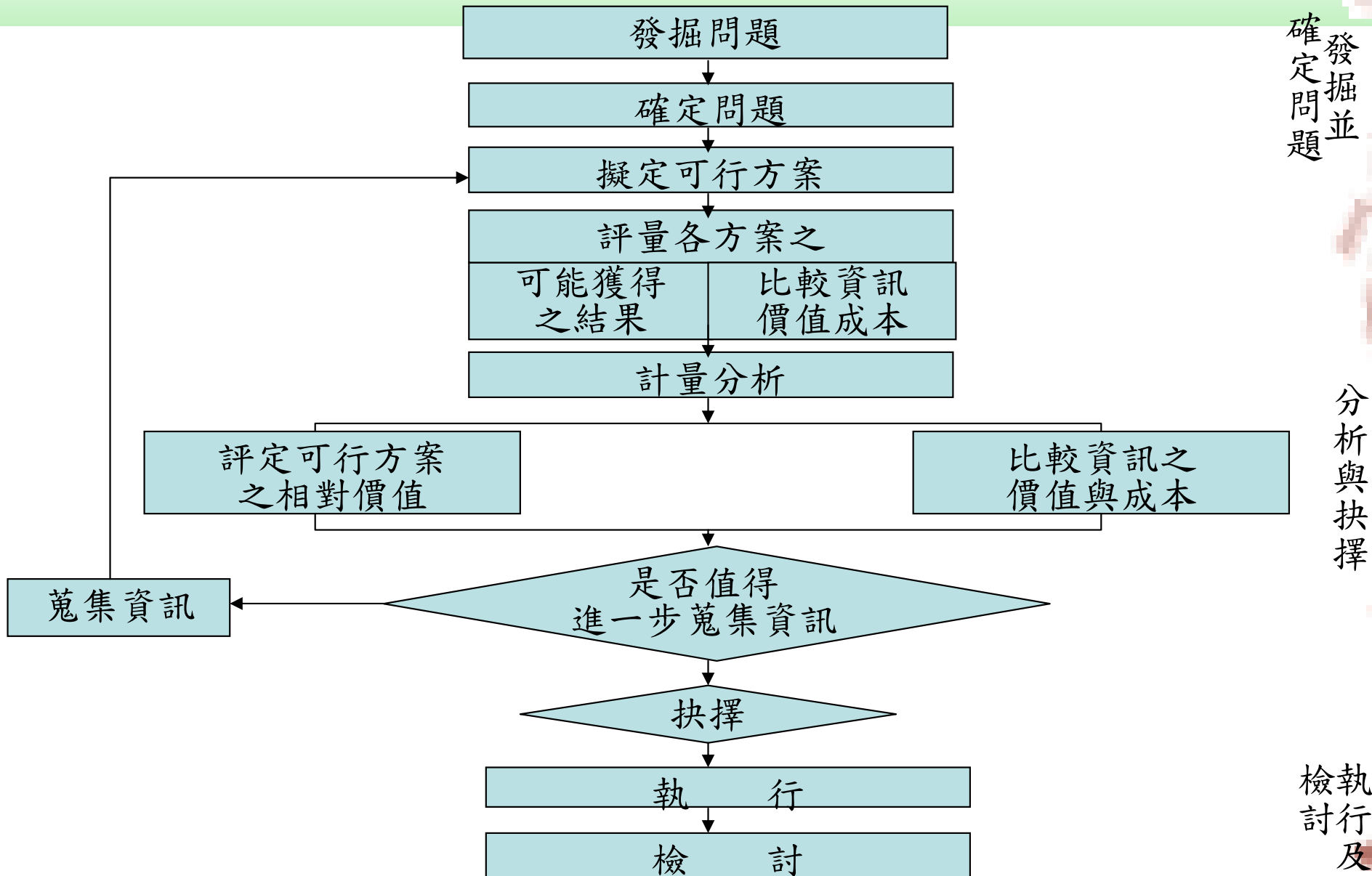
經驗

- 經由嘗試及接觸，不斷學習及累積未來處理問題應具之反應能力，避免相同錯誤之再發生。

系統思考

- 由於問題具複雜度，因此以邏輯性的程序分解問題，逐步考量相關因素，求得最佳方案。

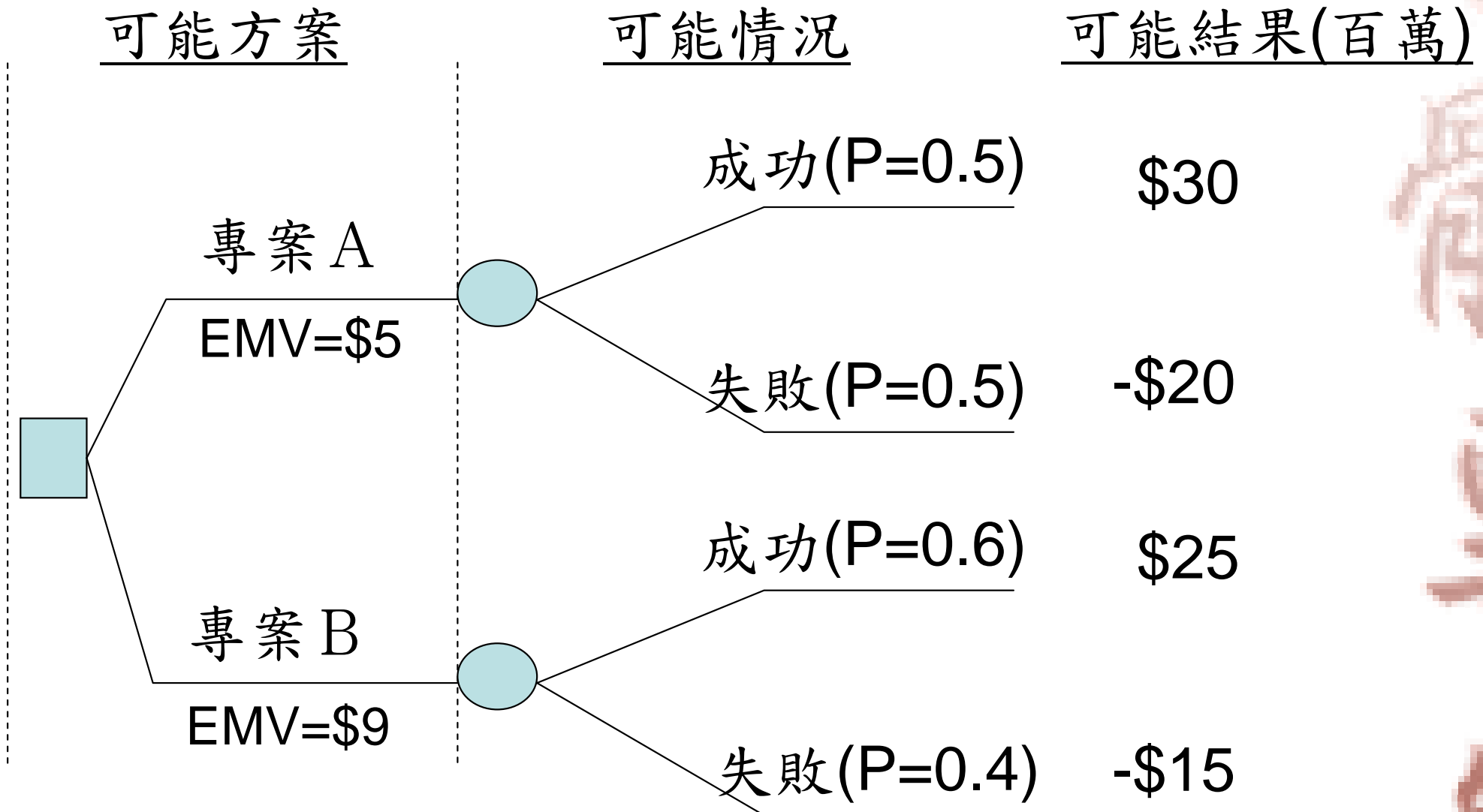
理性決策過程



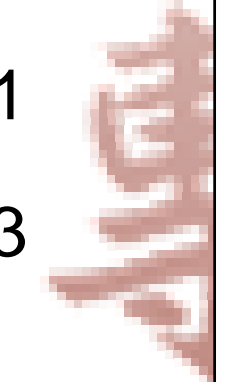
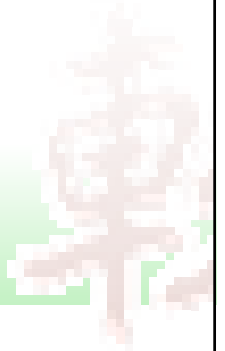
專案選擇方法

- ❖ 決策樹法
- ❖ 德爾菲法 (Delphi)
- ❖ 簡單計算法
- ❖ 速成演算法
- ❖ 決策表法
- ❖ 分析層級程序法 (AHP)
- ❖ 最大最小法則
- ❖ 成本效益法
- ❖

決策樹法



速成演算法



	專案A	專案B	專案C	專案D		小計	比序
專案A		0	0	0	➡	0	4
專案B	1		0	1	➡	2	2
專案C	1	1		1	➡	3	1
專案D	1	0	0		➡	1	3

綜合評量：C>B>D>A

簡單計算法

某項資訊系統建置案，經評估可行方案計A及B兩案

	A	B
成本分析	5	3
計畫風險	3	5
工作進行	6	4
整體規劃	3	7
總分	17	19

加權評估法決策表(1)

評量因素	權數 (1~10)	專案A(1~10)		專案B(1~10)		專案C(1~10)	
		得分	加權 分數	得分	加權 分數	得分	加權 分數
對本公司 商譽影響							
獲利率							
專案成功率							
公司資源是 否可以支援							
總分							

權數設定基準

W	重要程度
10	Extremely important
9	Highly important
8	Very important
7	Important
6	Fairly important
5	Probably important
4	Some important
3	A little important
2	Very little important
1	Unimportant

評分尺度

程度	內容
10	很好
8	好
6	稍好
4	稍差
2	差
0	很差

加權評估法決策表(2)

評量因素	權數 (1~10)	專案A(1~10)		專案B(1~10)		專案C(1~10)	
		得分	加權 分數	得分	加權 分數	得分	加權 分數
對本公司 商譽影響		5		9		4	
獲利率		7		6		8	
專案成功率		6		7		8	
公司資源是 否可以支援		3		4		7	
總分		21		26		27	

加權評估法決策表(3)

評量因素	權數 (1~10)	專案A(1~10)		專案B(1~10)		專案C(1~10)	
		得分	加權 分數	得分	加權 分數	得分	加權 分數
對本公司 商譽影響	8	5		9		4	
獲利率	6	7		6		8	
專案成功率	8	6		7		8	
公司資源是 否可以支援	3	3		4		7	
總分							

加權評估法決策表(4)

評量因素	權數 (1~10)	專案A(1~10)		專案B(1~10)		專案C(1~10)	
		得分	加權 分數	得分	加權 分數	得分	加權 分數
對本公司 商譽影響	8	5	40	9	72	4	32
獲利率	6	7	42	6	36	8	48
專案成功率	8	6	48	7	56	8	64
公司資源是 否可以支援	3	3	9	4	12	7	21
總分		139		176		165	

專案評選之基本模式－非計量法 (Nonnumeric or Qualitative) (1/2)

Qualitative

- Sacred Cow
- Operating Necessity
- Competitive Necessity
- Product Line Extension
- Comparative Benefit

Descriptive

- Policy Capturing
- Decision Modeling

Survey/Interview Techniques

Profitability

- Payback Period
- Average Rate of Return
- Discounted Cash Flow
- Internal Rate of Return
- Profitability Index

專案評選之基本模式－非計量法 (Nonnumeric or Qualitative) (2/2)

■ Scoring

- Unweighted 0-1 factor Scoring
- Unweighted factor Scoring
- Weighted factor Scoring
- Constrained Weighted factor Scoring

■ Other

(Normative/Explicit)

- Cost/Benefit Analysis
- Operations Research Models
- Explicit Linear Models

專案範圍定義

- 依產品架構發展工作分解結構 (Work Breakdown Structure, WBS)
- 界定工作產品
- 界定必須對外採購的工作產品
- 界定將要再使用的工作產品

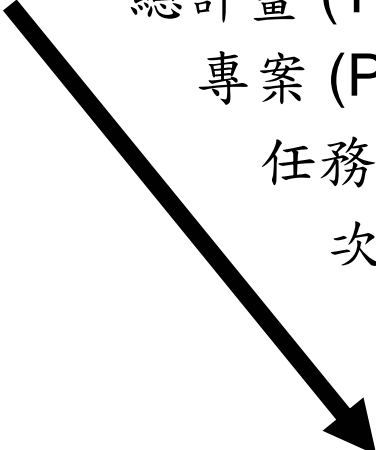
WBS基本觀念與定義 (1/4)

- 用任務導向活動群組（task-oriented “family tree”）的系統化方式，以組織、定義、分類及圖示專案所需執行的各工作項目以達成其目標。
- WBS可視為是一種專案「科層化」的結構圖形，用以區分專案不同層次的工作。
 - 專案可分解為任務（task）或活動（activities）、次任務（subtasks）、工作分項（work package）及工作要素（levels of effort）。
 - 每一項工作都是單一、有意義的或是由一群次級工作的組合。
 - 每一工作項目都被指派個別作業人力與職掌，且均被付予預算、估算工時及分配相關資源。

WBS基本觀念與定義 (2/4)

- 它可用不同型式的圖或表來顯示，六層結構式乃最普遍之運用方法：

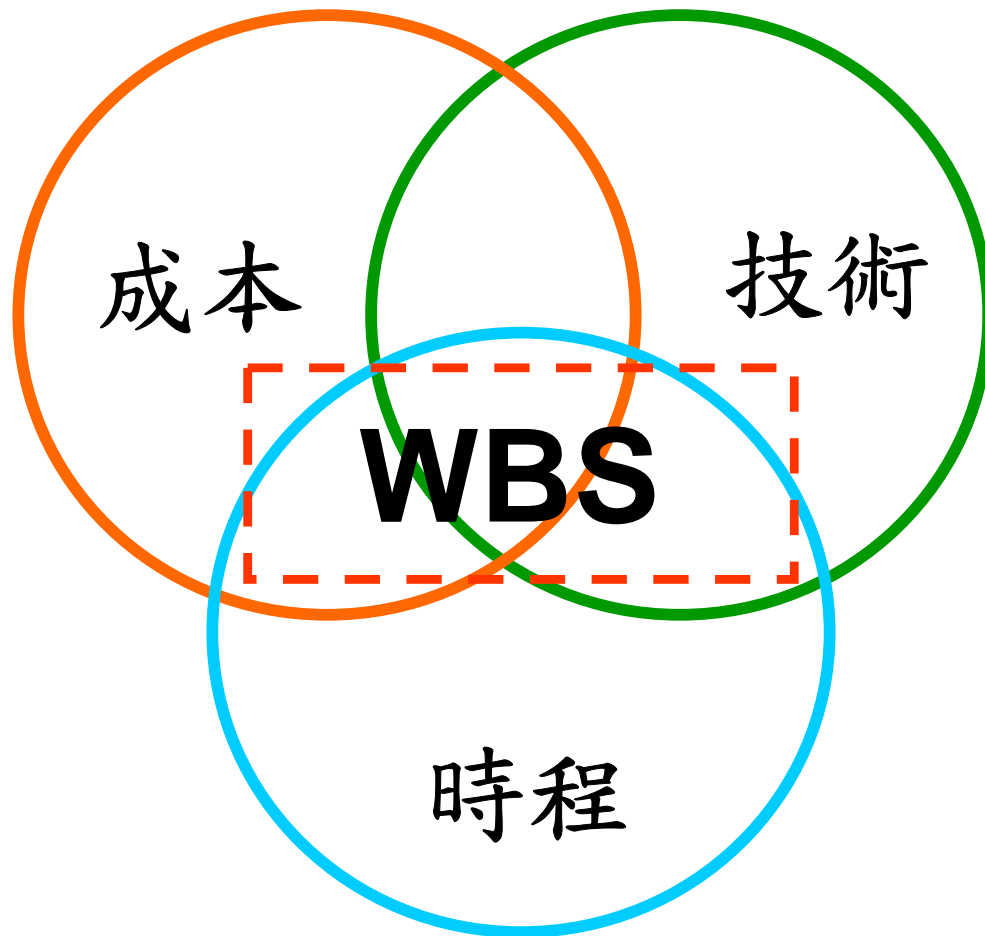
<u>層級 (Level)</u>	<u>描述 (Description)</u>
1	總計畫 (Total Program)
2	專案 (Project)
3	任務 (Task/ Activity)
4	次任務 (Subtask)
5	工作分項 (Work Package)
6	工作要素 (Level of Effort)



WBS基本觀念與定義 (3/4)

- 亦可為一樹狀架構的組合產品，以作為計畫管理、工作規劃、資源分配、及任務賦與活動之依據。
- 就工程施工作業言，完整之工作分解結構可有效管制承包商工程進行。
- WBS是準備及運用許多管理工具的基礎：網路、方塊圖（bar chart）、矩陣責任表、計價、風險分析等。

WBS基本觀念與定義 (4/4)



WBS 對專案經理而言是專案運作的核心，它可作為整合成本、技術及時程等三大限制因素的工具。

WBS之功能

- 將需執行之工作有系統的分類與組合，避免遺漏。
- 提供建立規格、時程、資源等需求之架構。
- 提供專案進度審查更有效的控制基礎。
- 提供研發專案管理各項活動（任務指派、預算編列、技術性規劃、計畫管理等）之依據。
- 為建立專案管理資訊系統之主軸。
- 提供明確的方法，以協助辨識合約交運標的及主要工作執行的決策點。
- 是確認所有必要工作是否已完成的重要依據。

WBS 製作的考量

- 對PM而言，WBS的製作是其最困難的任務之一，它是專案執行的核心程序，它之所以重要是因為：
 - 當解讀SOW、目標及規範內容時，主要的管理性及技術性的問題必須有所取捨。
 - 若干主觀的判斷會影響WBS的製作，尤其不同的單位或專業的人對WBS會有相異的意見。
 - 然而，無論有多少不同的版本專案計畫書只能接受一種，且必須是大家的共識。
- WBS的製作通常是很耗時的，當然仍需視專案大小與複雜程度而定。

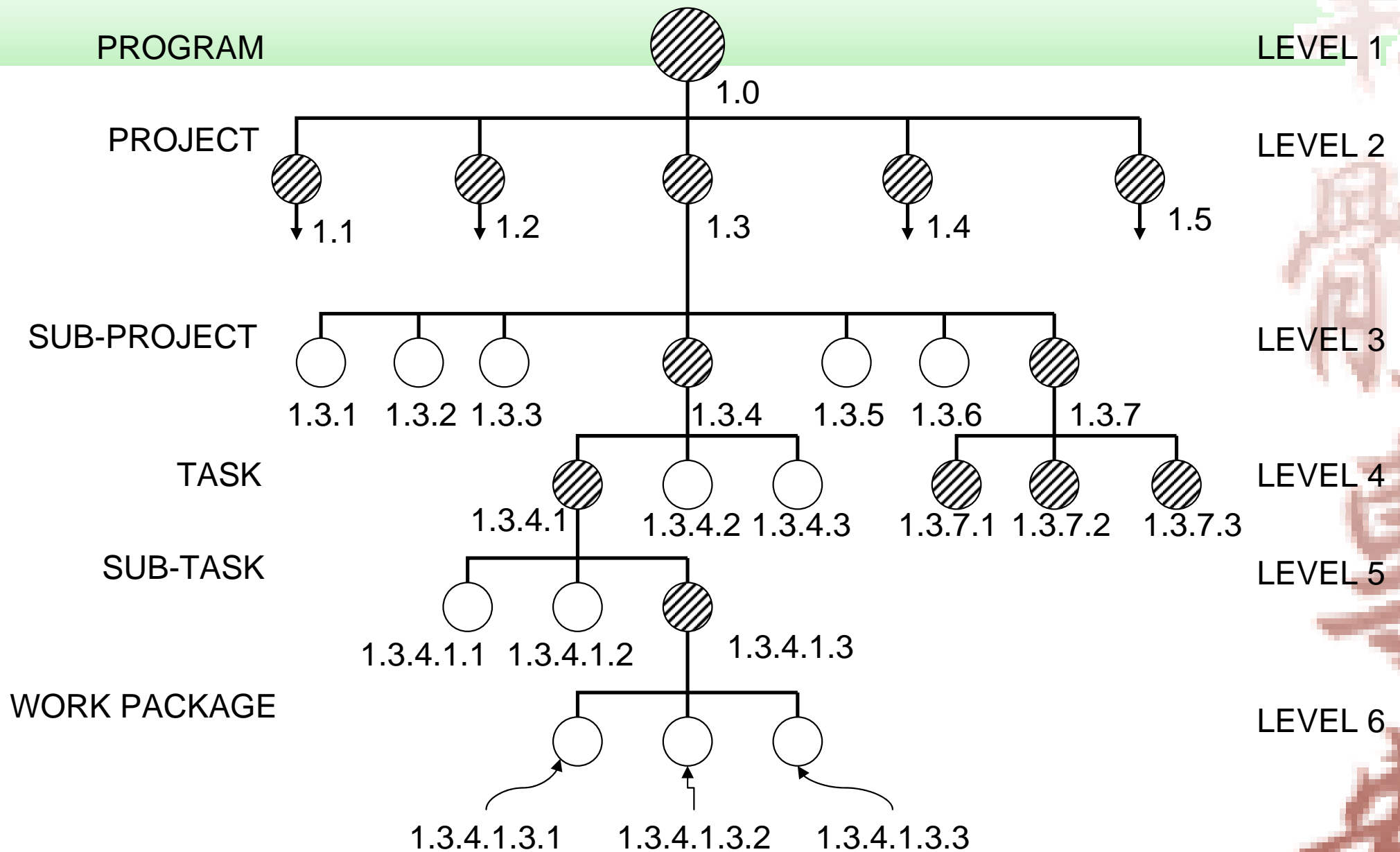
WBS製作的方法與步驟 (1/2)

- 當開始準備WBS時，可將其視為達成專案目標而必須執行的工作檢查清單（checklist）。
- 用一個結構性、系統性的方式來處理該工作檢查清單，如：
 - 首先將專案區分為幾個主要的部份、或類別、階段、或若干專案、系統。
 - 其次，再把每一部份分解成最小而不可分割的單位（如任務、次任務、工作分項）。
 - 最後再按層次，將每一分解出的工作分項逐次組合在不同的系統中。

WBS製作的方法與步驟 (2/2)

- 腦力激盪、分組會議及審閱過去類似專案記錄為準備WBS的主要方法。
- 當確信已無任何主要事項被遺忘或漏列時，WBS的製作方可結束。
- 在專案的任務被已分解至可確認的工作分項，則需針對其回答下列問題：
 - 所需之人力資源為何？
 - 誰應該負責這項工作？
 - 為完成任務還有什麼其他考量？
 - 什麼是需要的工作時程？

WBS製作的編號方式



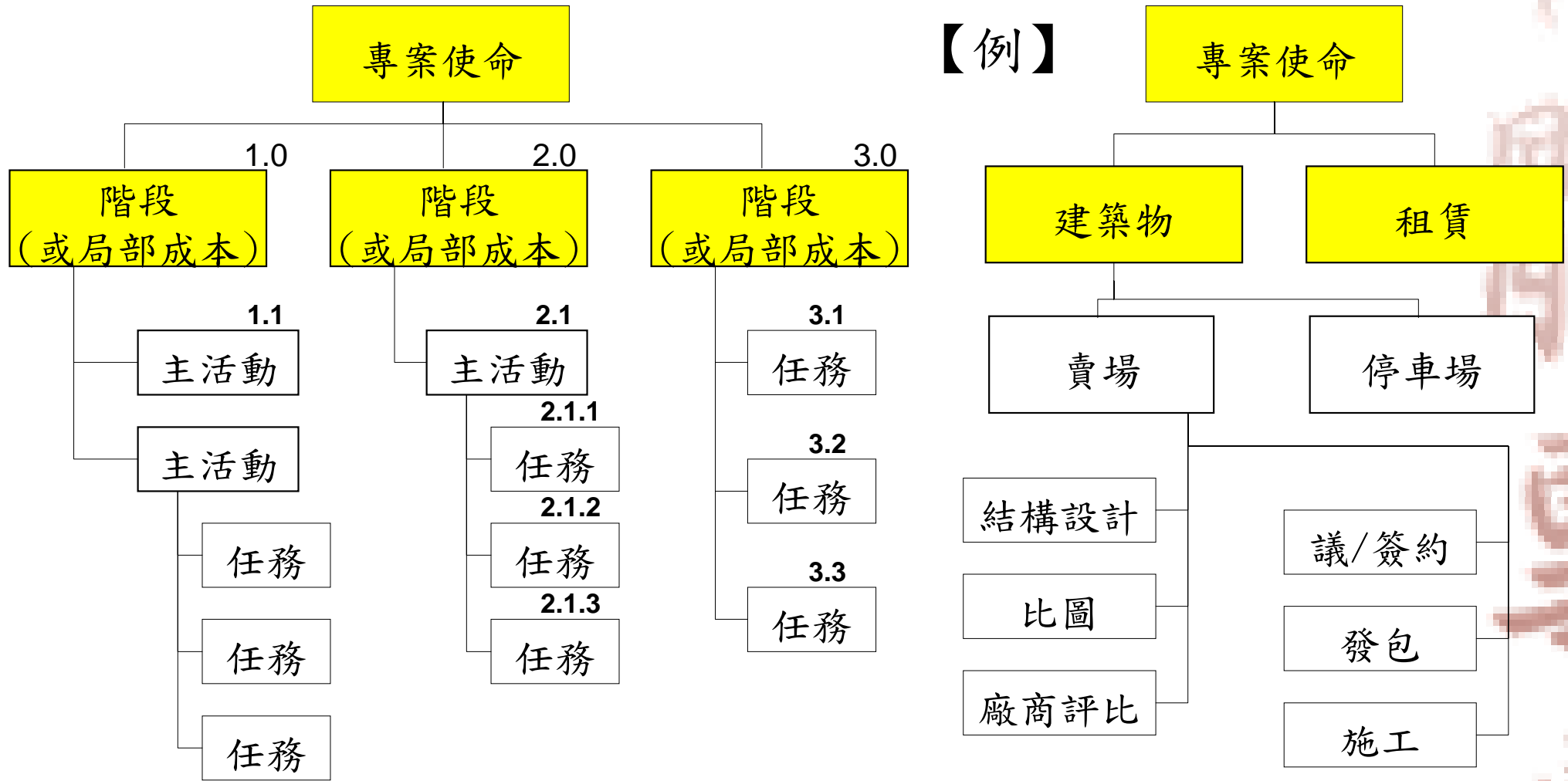
WBS製作的方法：工作分項內容 (Work Package Contents)

1. 專案抬頭 (Name of Program of Project)
2. WBS編號 (WBS No.)
3. 工作分項負責人 (Responsible Person)
4. 工作分項名稱 (Work Package Title)
5. 工作範疇 (Scope of Work)
6. 交運標的 (Deliverables)
7. 時程 (Schedule)
8. 前一工作事項 (Predecessor Events)
9. 接續工作事項 (Successor Event)
10. 風險程度 (Risk Level)

WBS製作的原則

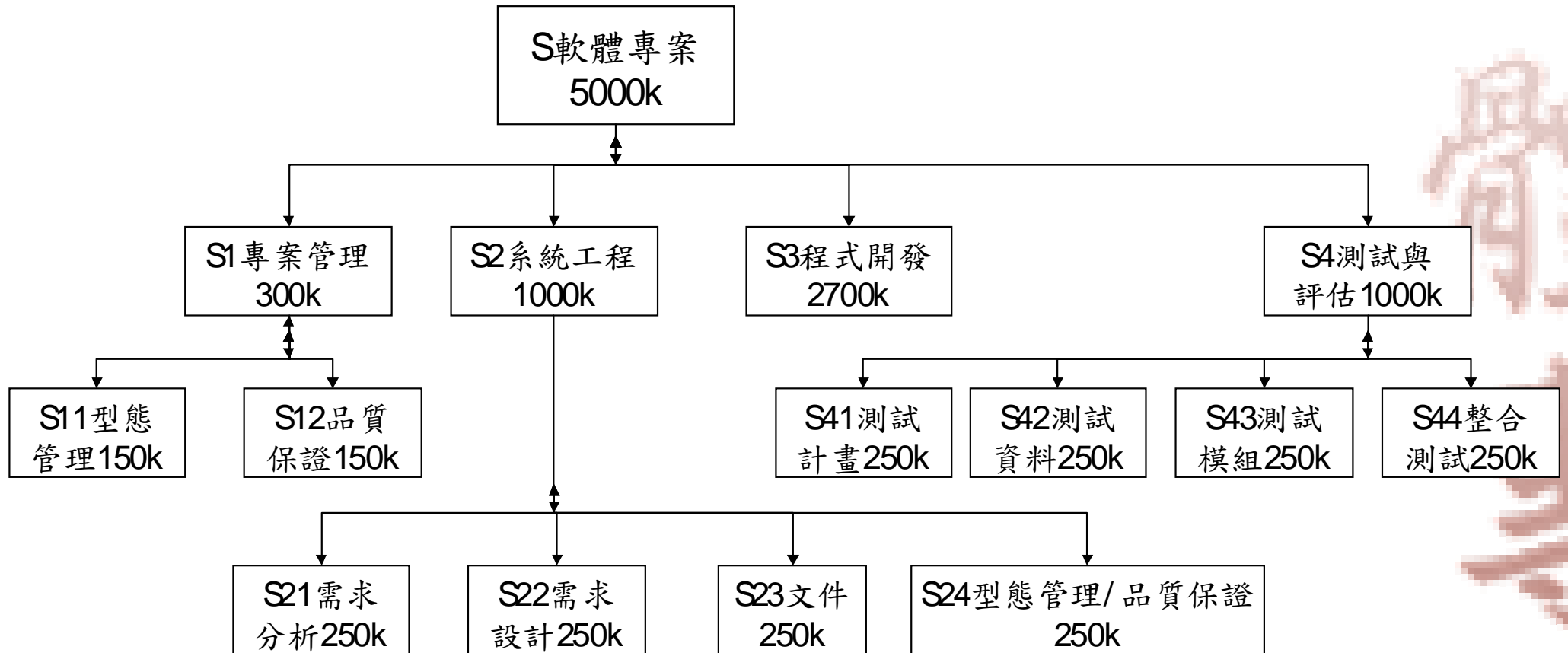
- 工作分解結構代表專案目標之達成，及產品品質規格。
- 工作分解結構應集合各專長領域人員共同完成，並應反覆檢查無缺漏。
- 工作分解結構至多不宜超過七層（美國政府要求合約商提供WBS到第三層）。
- 數字編碼系統應運用於結構中各工作，以利管制進度及資訊化運用。
- 各工作項目應有具體之產品產生。
- 各工作項目及其活動應具獨立性。

WBS系統性架構



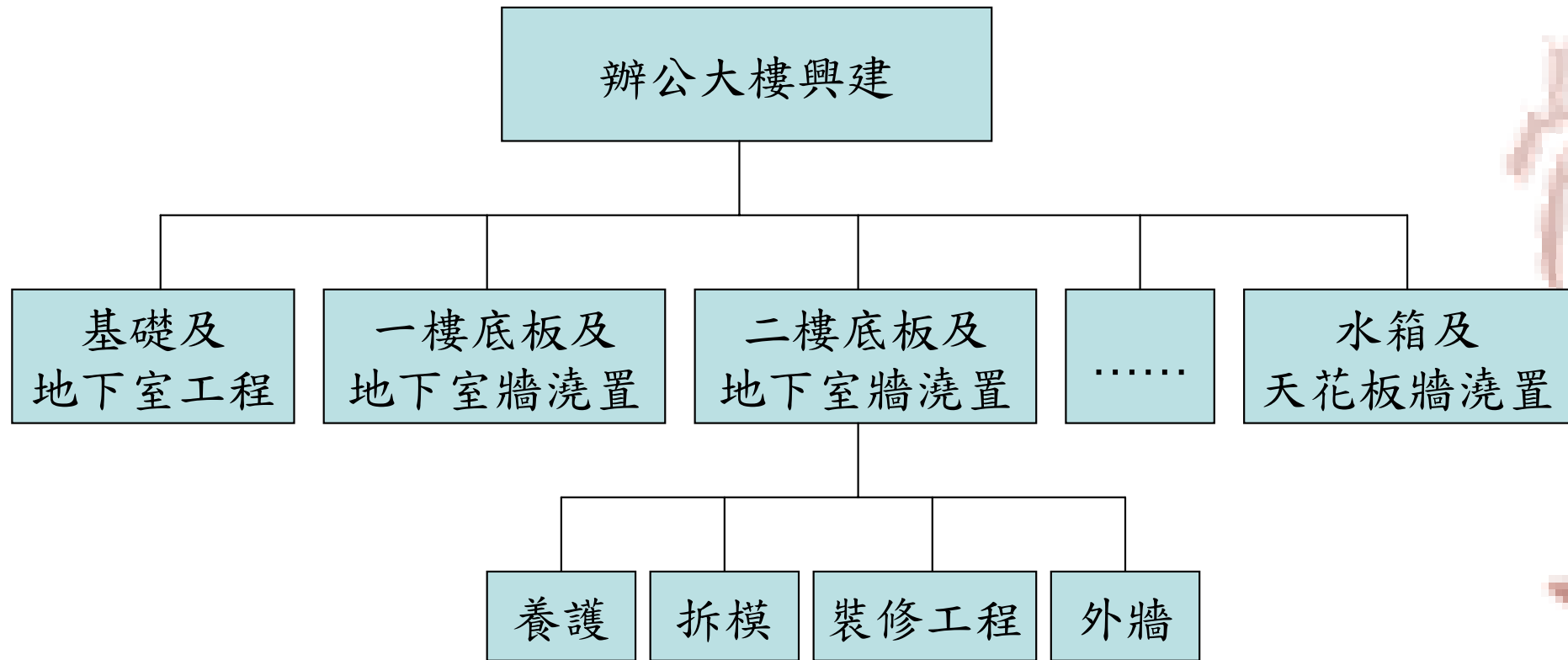
管理層次 (managerial or outcome levels)
 技術層次 (technical or process levels)

軟體專案WBS

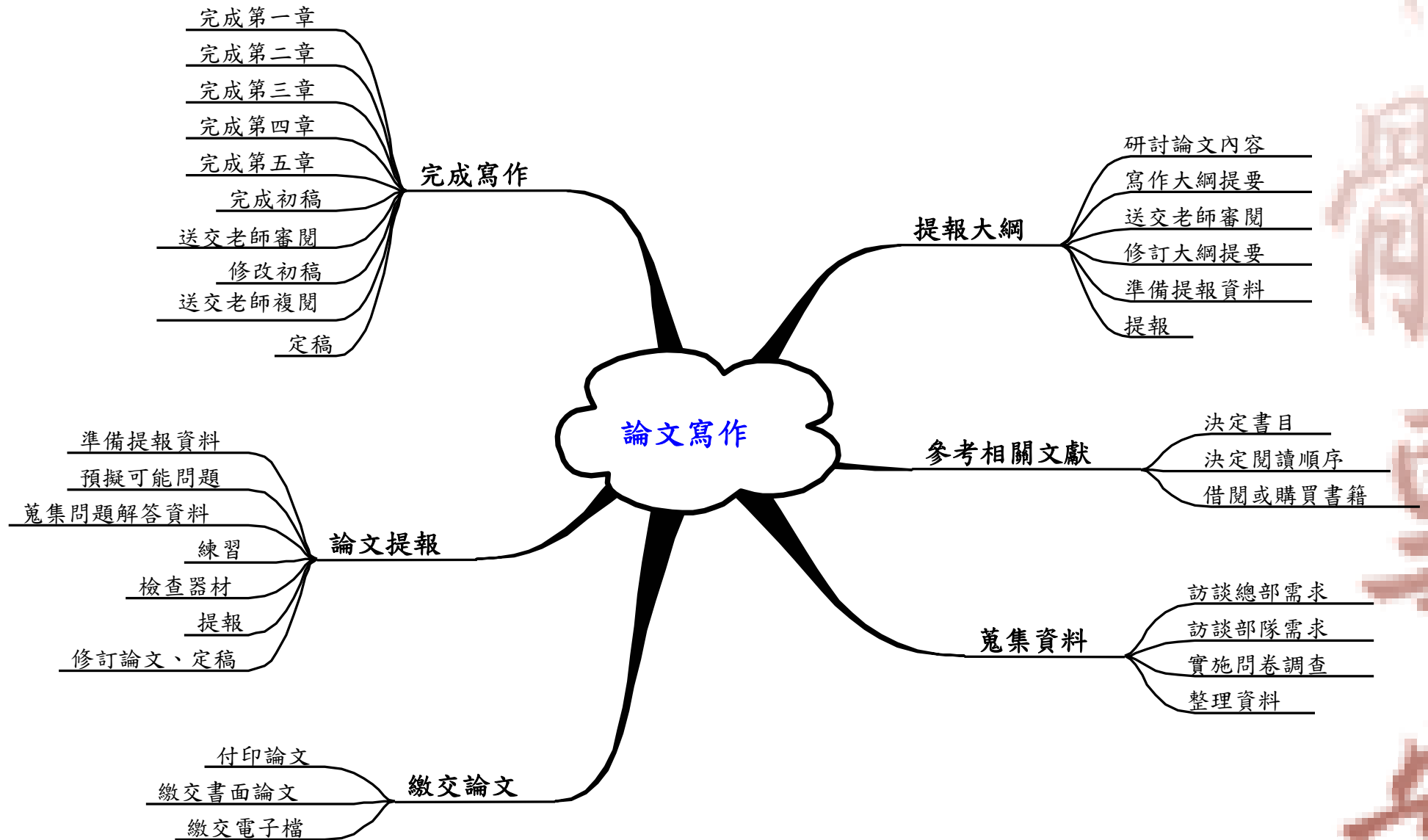


資料來源：「軟體專案管理 Software Project Management」(林信惠·黃明祥·王文良著)

WBS 任務之分解：按工作流程



論文寫作WBS：水平樹式



條列清單式：軟體發展WBS

1.2.3 Software Development

1.2.3.1 Management

1.2.3.1.1 Project

1.2.3.1.1.1 Planning

1.2.3.1.1.2 Administration

1.2.3.1.2 Subcontract

1.2.3.1.2.1 Planning

1.2.3.1.2.2 Administration

1.2.3.2 Design

1.2.3.2.1 Functional Specification

1.2.3.2.2 Interface Specification

1.2.3.2.3 Database Specification

1.2.3.2.4 Subsystem Design
Specification

1.2.3.3 Build (Code & Unit Test)

1.2.3.4 Integration

1.2.3.4.1 Planning

1.2.3.4.2 Software Integration

1.2.3.4.3 Hardware Integration

界定工作產品

- 須詳細到足以作專案工作項目估計、責任指派及時程安排。
- 藉由最高階的WBS協助，以工作項目、組織角色與責任來判斷專案工作份量。
- 除開發外，特別注意測試及審查之工件量。

Karl Wiegers的軟體需求層級

企業需求

企業組織或與其有利益關係的客戶，對於所進行的專案（系統）或產品所要求的高階目標，包括企業對於專案的願景（vision）與範圍（scope）。

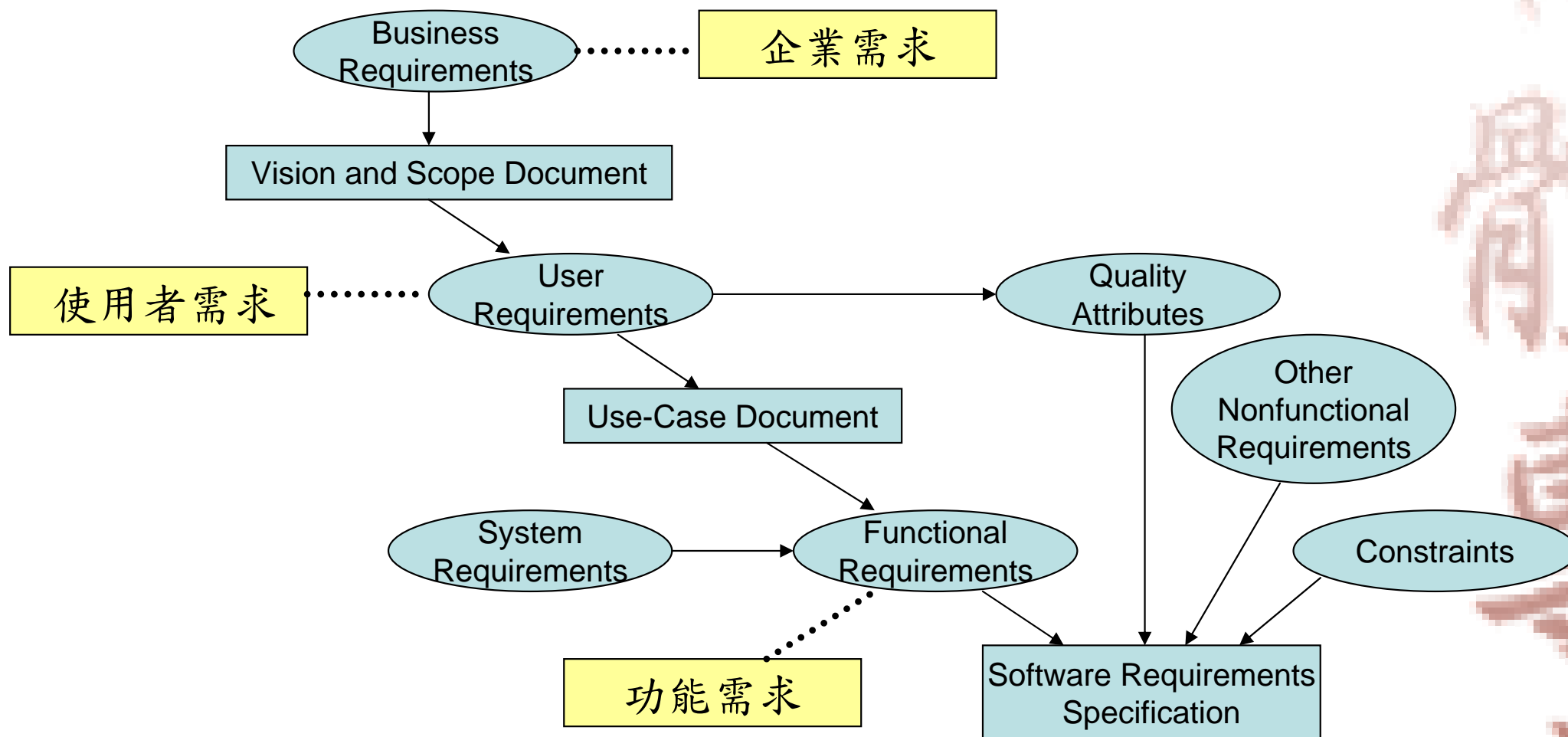
使用者需求

指實際使用產品的使用者，其所預期的產品功能，可用使用者案例（use case）來描述。

功能需求

開發者必須建構的產品功能，可讓使用者完成工作已滿足企業需求。功能需求經歸檔整理成軟體需求規格（SRS）。

軟體需求層級及各元件的關係圖



資料來源：Software Requirement by Karl E. Wiegars

軟體需求架構與需求管理模式

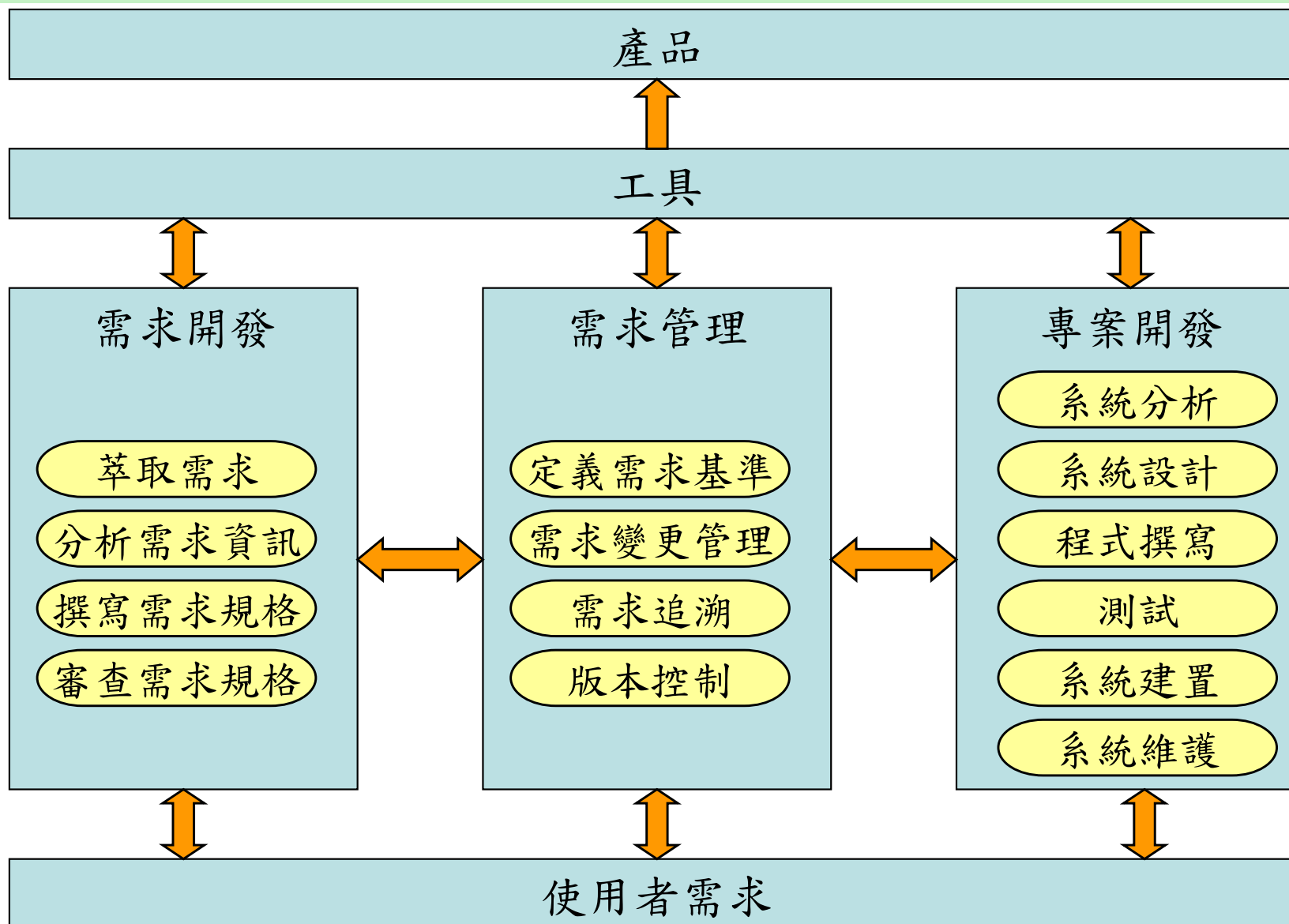
■ 軟體需求架構

- 描述資訊系統開發過程中軟體需求的運作架構，以使用者需求為資訊來源，並透過需求開發、需求管理與專案開發等資訊系統開發過程，加上個別輔助工具，完成最終產品。

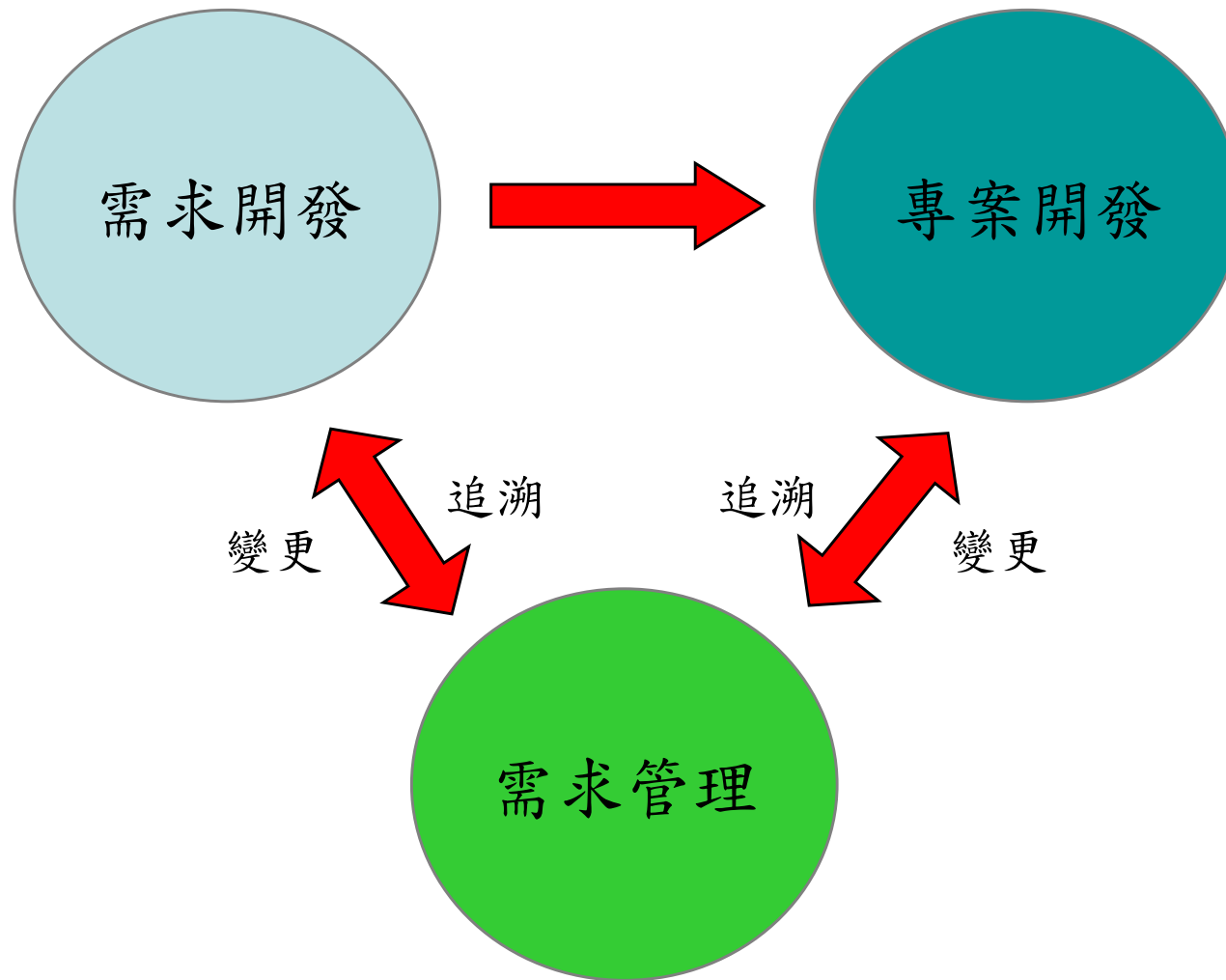
■ 需求管理模式

- 本模式的主要核心部份在探討需求管理的工作流程。核心工作包括「定義需求基準」、「需求變更管理」、「需求追溯」及「版本控制」等。

軟體需求架構



軟體發展流程關係圖



Requirements Development

■ Purpose:

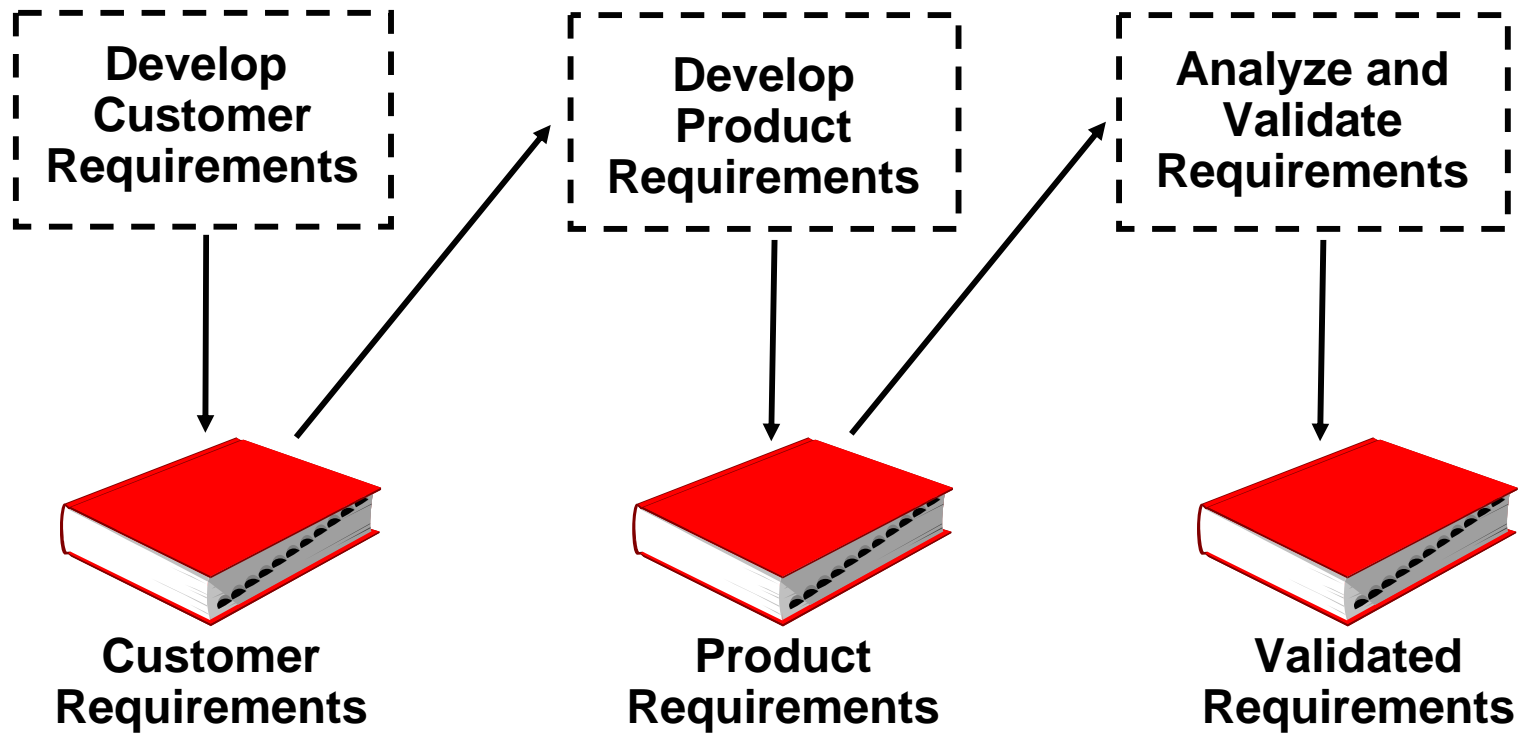
- Produce and analyze customer, product, and product component requirements.

需求開發的工作項目

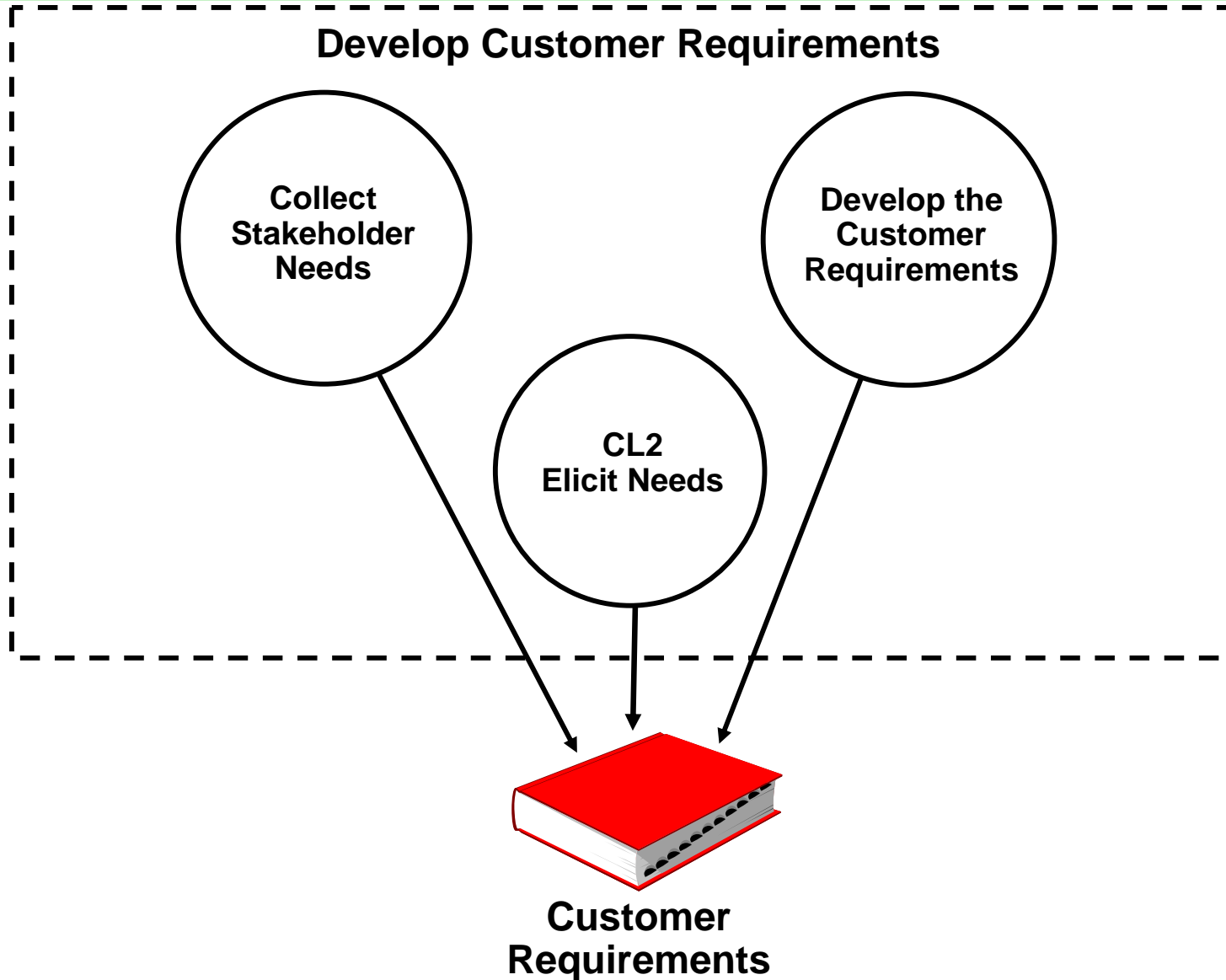
Karl Wiegers (1998)

- 指出產品的預期使用群。
- 從各個使用群代表取得需求。
- 了解使用者的工作及目標，以及這些工作要達成的企業需求。
- 分析取自使用者的資訊，區分解決工作需求的背後的功能需求、企業規範、品質特性、建議的解決方案，以及額外資訊。
- 將系統層級的需求分割為子系統功能，並將這些需求加入軟體元件。
- 了解品質特性間關聯的重要性。
- 協商需求的優先順序。
- 將蒐集的使用者需求加以詮釋，撰寫成規格及模型。
- 審查需求規格，以確定對使用者提出的需求達到共識，並於開發團隊採用前修正任何問題。

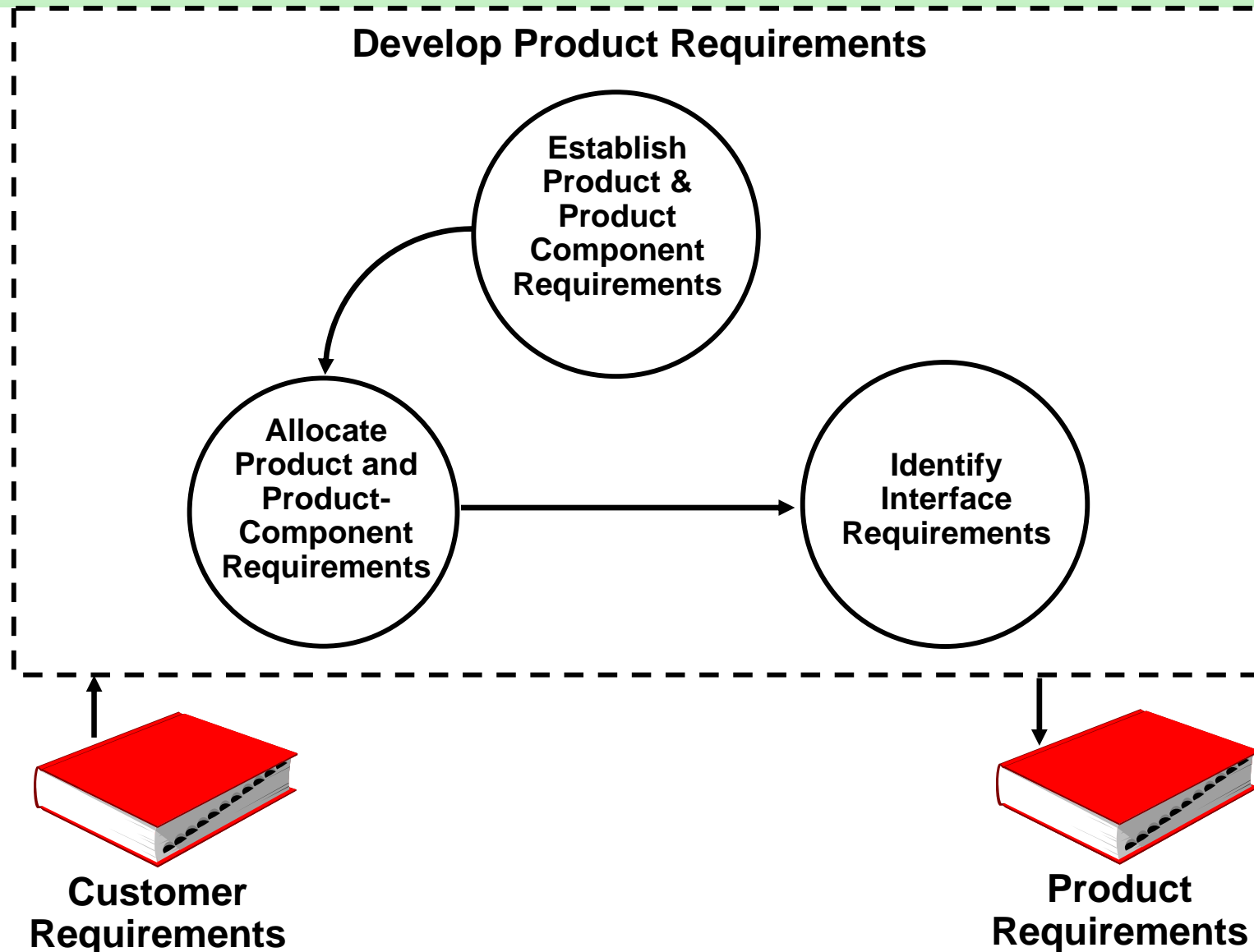
需求開發的內容 (1/4)



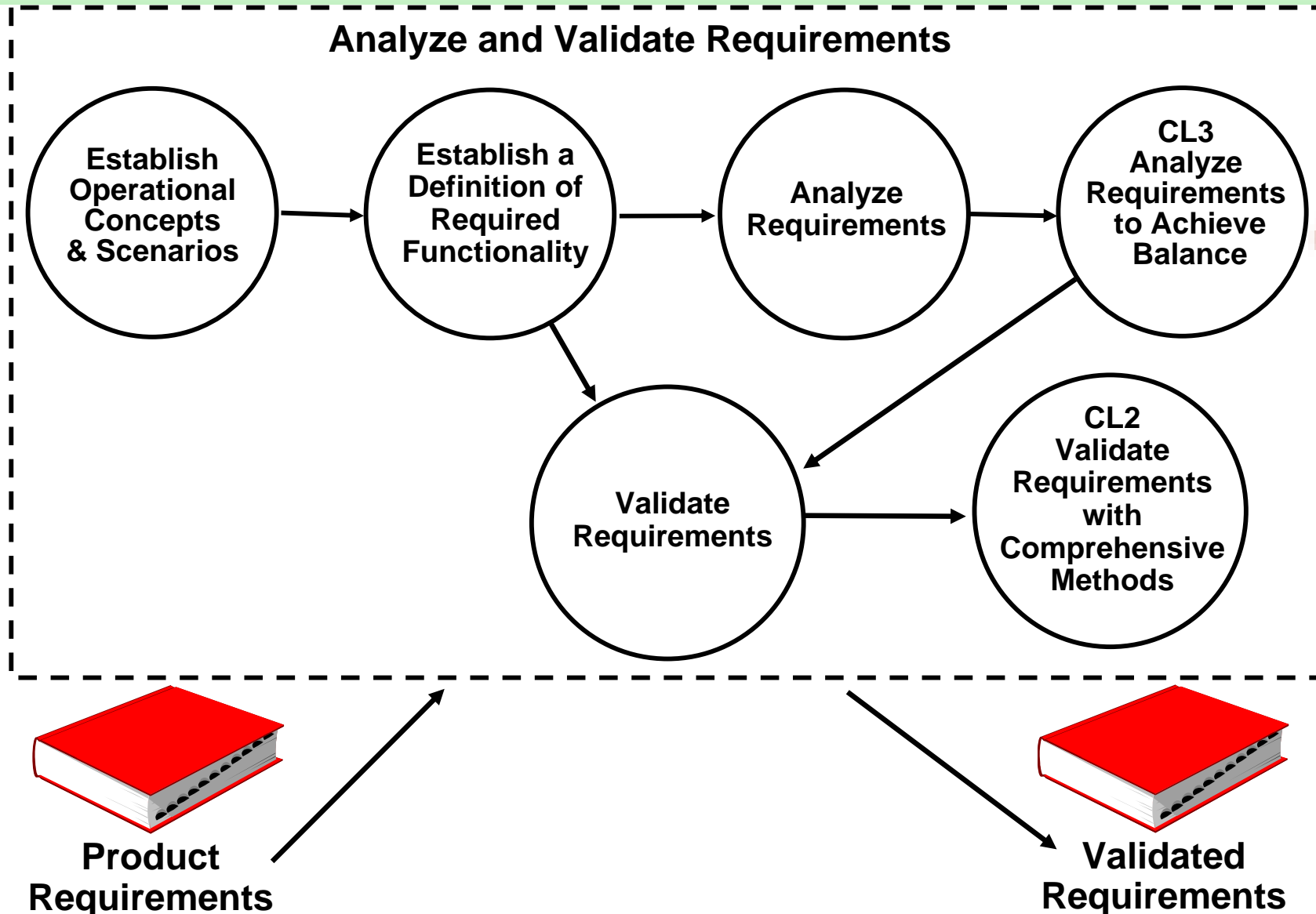
需求開發的內容 (2/4)



需求開發的內容 (3/4)



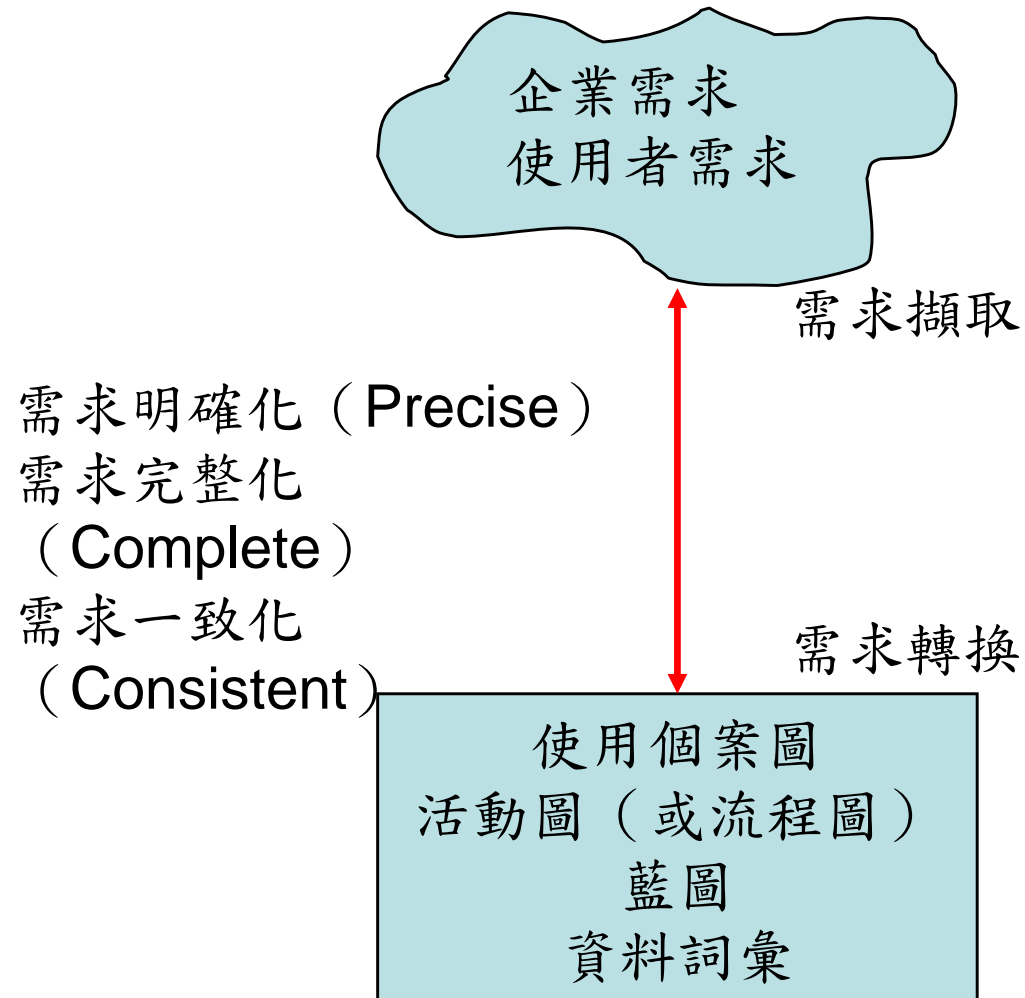
需求開發的內容 (4/4)



需求萃取的實施

- 利用「需求建模」的方式，將客戶「意識化之需求」具體表達出來。
- 採用物件導向的觀念，使用UML塑模方法，從問題領域探討如何萃取出客戶的需求，以及如何使用案例圖表示出系統所需的功能、系統的使用者與外部系統。

需求建模



資料來源：吳仁和、林信惠－系統分析與設計

需求管理

目的

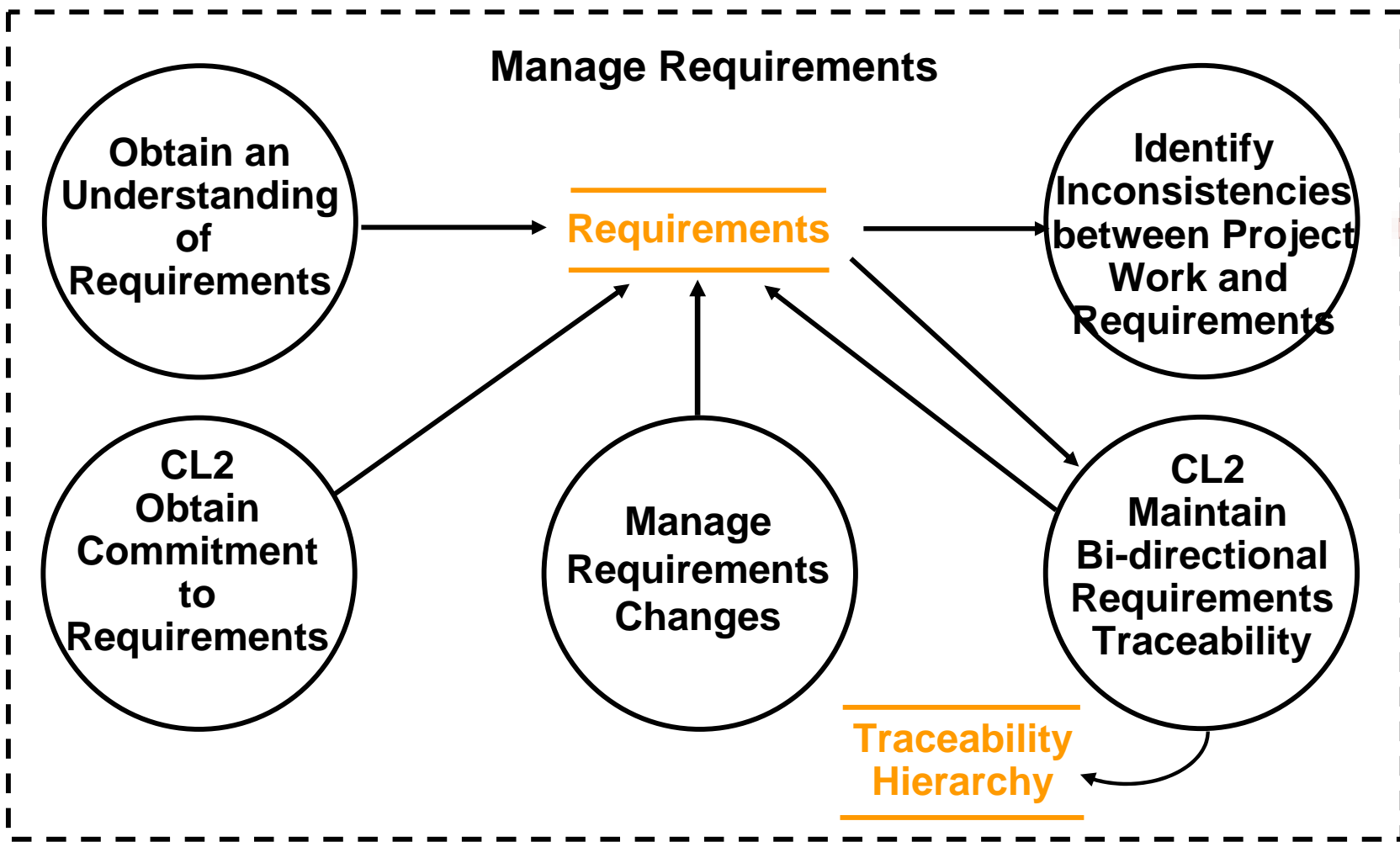
- Manage the requirements of the project's product and product components.
- Identify inconsistencies between those requirements and the project's plans and work products.

需求管理

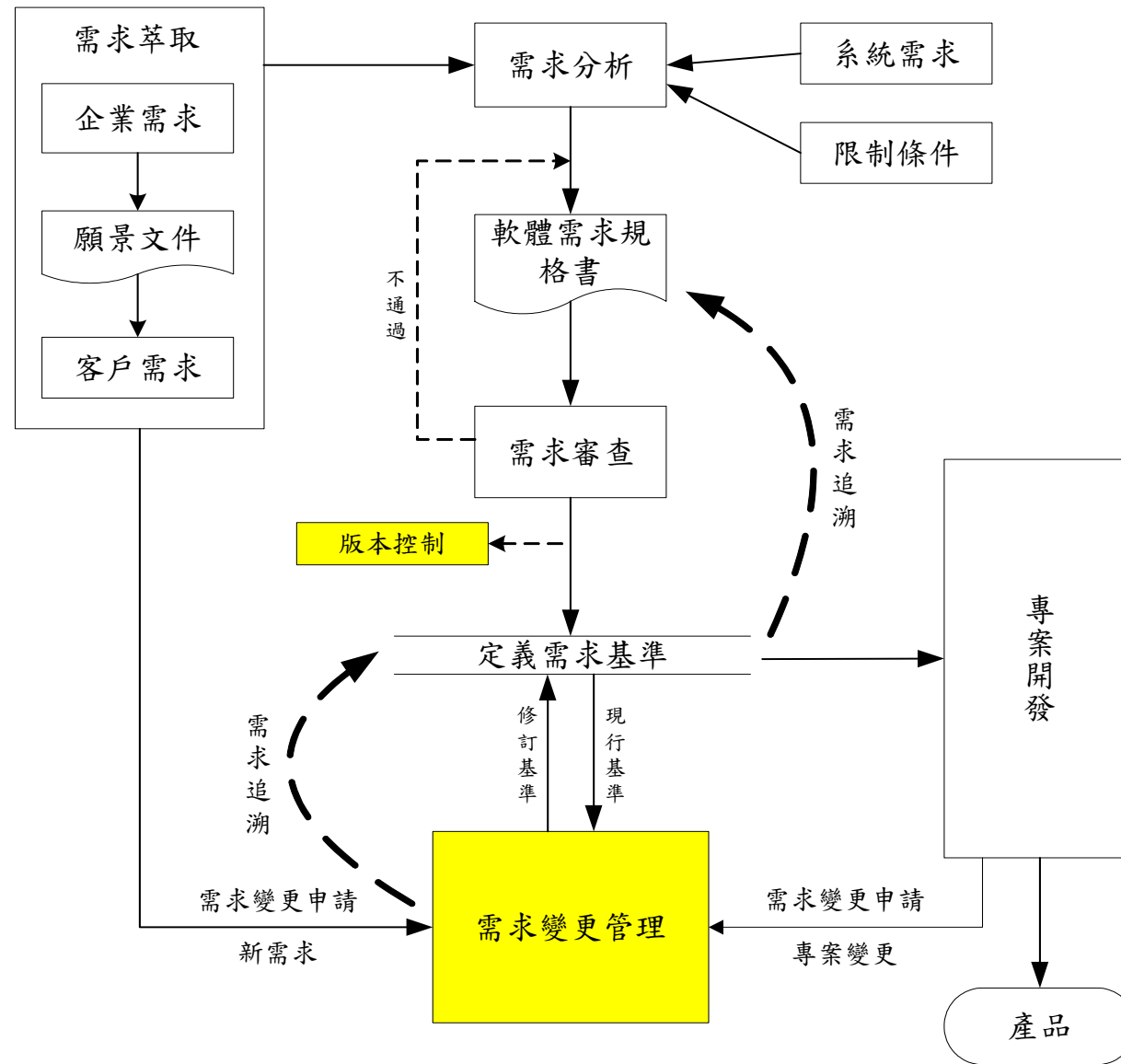
工作項目 (Karl Wiegers, 1998)

- 定義需求的基準（適時提出摘要以代表目前同意的需求）。
- 審查需求的變更申請，評估其衝擊後再決定是否採用。
- 在控制下將同意的需求變更整合至專案。
- 使專案規劃與需求同步進行。
- 依照需求變更的評估衝擊，進行溝通。
- 追蹤個別需求的相關設計、原始程式碼及測試案例。
- 於專案進行中追蹤需求狀態及變更。

需求管理的內容



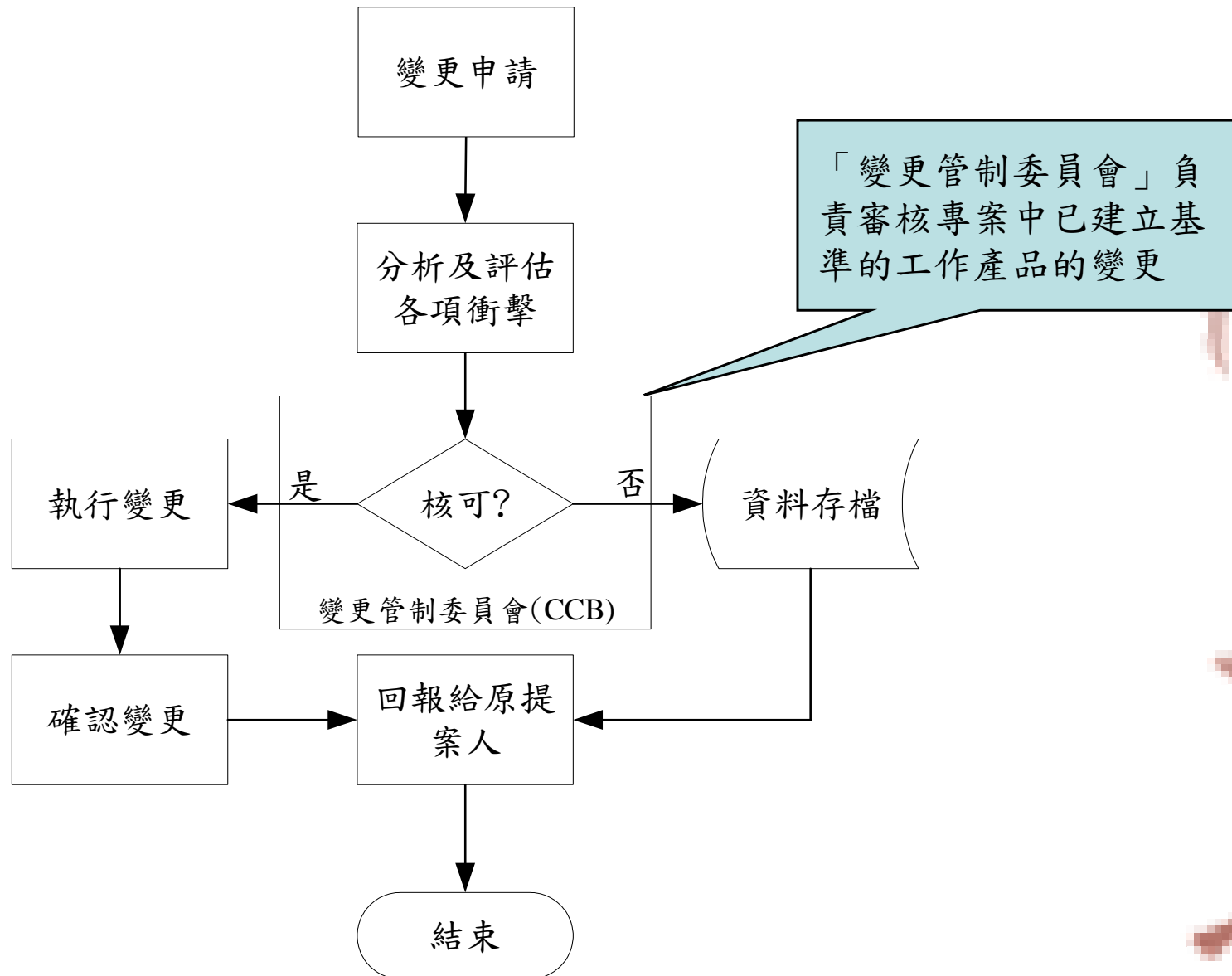
需求管理模式



需求變更管理的原則

- 所有需求變更必須依照流程進行。如果變更要求並非依照流程提出，就不應納入考慮。
- 變更未經核准前，除非經過可行性探索，否則均不可進行設計或採用。
- 僅提出要求變更並不代表會落實，應該由專案變更管制委員會（Change Control Board, CCB）裁定是否採用。
- 變更資料庫的內容必須開放給所有專案相關人員閱讀。
- 每個納入的需求變更必須能透過已核准的變更要求追蹤。

需求變更管理流程



需求變更管理的工具

- 需求管理的工作十分繁瑣，透過需求管理模式，固然可以了解需求的管理流程，但對於個別需求的狀態追蹤，如果不借助需求管理工具的幫忙，將需求相關資訊儲存在資料庫，而僅利用手邊的簡易工具，實際上是很難做到有效的管理。
- 採用IBM Rational RequisitePro需求管理工具，它結合了微軟Word和需求資料庫，使用戶在實際軟體發展生命週期內，能夠輕鬆建立需求之間的關聯，管理需求的變更。

Configuration Management

■ Purpose:

- Establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits.

Configuration Management - Context

